

Sparse Representation of a Blur Kernel for Blind Image Restoration

Chia-Chen Lee, and Wen-Liang Hwang

Institute of Information Science, Academia Sinica, Taiwan

Abstract—Blind image restoration is a non-convex problem which involves restoration of images from an unknown blur kernel. The factors affecting the performance of this restoration are how much prior information about an image and a blur kernel are provided and what algorithm is used to perform the restoration task. Prior information on images is often employed to restore the sharpness of the edges of an image. By contrast, no consensus is still present regarding what prior information to use in restoring from a blur kernel due to complex image blurring processes. In this paper, we propose modelling of a blur kernel as a sparse linear combinations of basic 2-D patterns. Our approach has a competitive edge over the existing blur kernel modelling methods because our method has the flexibility to customize the dictionary design, which makes it well-adaptive to a variety of applications. As a demonstration, we construct a dictionary formed by basic patterns derived from the Kronecker product of Gaussian sequences. We also compare our results with those derived by other state-of-the-art methods, in terms of peak signal to noise ratio (PSNR).

Index Terms—IEEEtran, journal, L^AT_EX, paper, template.

I. INTRODUCTION

IMAGE blurring is a common problem in digital imaging generally when pictures are taken with wrong focal length, camera shake, object motion, or shallow depth of field, to name a few [1], [2]. In this regards, blind image restoration problem involves restoration of image X from a noisy observation Y , which contains less information provided by blur kernel H :

$$Y = H *_2 X + N, \quad (1)$$

where N is noise, and $*_2$ indicates the 2-D convolution operation. However, the blind restoration method is difficult as it is impossible to restore X without a simultaneous restoration of H . Since the restorations of H and X are not jointly convex, the solution of blind restoration is highly dependent on the initial guess of H or X . This dependence can be lessened, but not completely removed, even through the imposition of convex regularizations on X and H .

Several attempts have been made to regularize image X , and all of them aimed to recover local high frequency components (edges) of X from image Y . The widely adopted total variation approach, which implicitly models X as a piecewise smooth function, intends to derive an image having a small total variation. Sparse representation models X as an element in a subspace which is spanned by a few unknown atoms in a dictionary. By contrast, there is no widely accepted regularization on the blur kernel H . Regularizations on H vary widely from a parametric form (a Gaussian function with unknown

standard deviation for handling out-of-focus blurring) to l_1 -norm on the support of H (for handling motion blurring). In this paper, we propose a novel approach by imposing on H a union of subspace models. In this approach, we assume that H can be sparsely represented over a dictionary of atoms. We believe that this approach can set a connection between the various forms of regularizations on H through the dictionary design. This approach will not only incorporate the parametric approach of imposition on H (through assumption of a dictionary of atoms, with each atom derived by certain parameter values), but it will also allow H freedom to adapt to various applications, because the dictionary can be either for general purpose or trained for a specific application.

Our mathematical model for blind image restoration is stated as follows, where images X and Y are converted to vectors x and y , respectively:

$$\begin{cases} \min_{H,x} \|y - Hx\|^2 + R(x) + \|\gamma\|_1 \\ H = \sum_i \gamma_i D_i. \end{cases} \quad (2)$$

where R is a convex regularization term on image, $D = [D_i]$ is a tensor dictionary (vector of matrices) composed of basic blur kernel D_i , and $\gamma = [\gamma_i]^T$ is a column vector of real numbers. The notation H is overloaded here with slightly different meanings in Equations (1) and (2).

The imposition of convex regularizations on x and γ facilitates an alternative approach to deriving the solution. Since each sub-problem, derived by fixing either X or H , is convex, the convergence of the approach is ensured. Nevertheless, the solution is still dependent on the initial guess of H or X . To note, modelling H as a sparse representation of D has an edge on guessing the initial H . If we have a dictionary of large number of atoms, a reasonable guess of H to start an algorithm is a function that will contain only one atom in the dictionary (this corresponds to approximation of H by its dominating atom).

In this paper, we demonstrate our approach by constructing a dictionary of blurring patterns formed by the Kronecker product of two 1-D Gaussian functions of various scales. Our dictionary assumes that the unknown blur kernel is sparse with respect to a mixture of out-of-focus Gaussian type blurriness. Although our dictionary is not generic for all types of blurriness, the construction process can be used to derive dictionaries for other types of blurriness. The initial guess of H is derived from estimation of the out-of-focus blurriness in images. The image is estimated by the variable splitting technique and the blur kernel is generated by the efficient proximal gradient method. We also demonstrate our restoration

results and compare the PSNR performance with other blind restoration approaches.

Notation.

Capital letters denote matrices and small letters denote vectors or scalars. The *vec* operation transfers a matrix into a vector by stacking one vector underneath the other. The inverse of *vec* is denoted as vec^{-1} . If a capital letter is used to denote an image, the corresponding small letter will denote the vector obtained by applying *vec* on the image. For example, if X is an image, then x is a vector with $x = \text{vec}(X)$. We use \otimes to denote the Kronecker product. Applying *vec* on both sides of $C = AXB$ yields

$$\text{vec}(C) = \text{vec}(AXB) = (B^T \otimes A)\text{vec}(X). \quad (3)$$

The rest of the paper is organized as follows. Section II reviews the related work. Section III formulates the blind restoration problem, models a blurring kernel as a sparse representation of a dictionary, and devises the procedure to construct the dictionary. In Section IV, an alternating minimization algorithm is proposed for blind restoration. Some important steps of the algorithm are also discussed. In Section V, we will report the tests conducted on monochrome and color image with various synthetic and real-life degradations and comparison of the results with other methods. Section VI contains our concluding remarks.

II. RELATED WORK

Numerous approaches have been proposed to remove blurriness on observed images under various circumstances (the deblurring problem). They can be roughly categorized, according to whether the blur kernel H is spatially variant as well as how much information of the kernel is provided. Some of the relevant studies have been listed in Table I and categorized according to the problem addressed. Although a large volume of work has been reported on deblurring problems, the current trend appears to shift from non-blind to blind category.

The main technical challenge to resolve non-blind cases is imposition of regularizations on images. On the other hand, the technical challenges for blind cases are the non-convexity of the problem, the determination of the blur kernel, and design of the regularizations on the blur kernel and the image.

Among all different forms of regularizations on images, the total variation (TV) regularization function and its variations have been widely used [3], [4], [6], [14], [15], [16], [17], [19], [8], [29], [30], [31], [20]. Statistical models on the gradients of natural images have been adopted for image prior [5], [6]. Sparse assumptions have been used to model representation coefficients for natural images in a transform domain [9], [8] or in an image domain [13]. A counter-intuitive finding is reported in [5] indicating that most cost functions for image prior prefer blurry images to sharp images, as blurry images have lower costs than sharp images. Attempt is made in [18] to achieve the lowest cost for true sharp image by introducing an l_1/l_2 function on images.

The spatially variant cases are solved by dividing an image into blocks (of size depending on the supports of blur kernels) and then processing each block independently by a spatially

invariant method. Computational complexity is one of the main concerns for the spatially variant methods, because fast Fourier transformation (FFT) can be applied only to spatially invariant cases but not to spatially variant ones. Some fast algorithms based on variable splitting techniques and proximal point methods have also been proposed [19], [8]. For example, the variable splitting technique is used in [19] for the TV approach and in [20] for statistical modelling of image gradients. Another topic of concern is the removal of boundary effects incurred from dividing an image into blocks [38] such as the matting approach adopted in [29].

As mentioned in [21], more the information of a blur kernel available for blind image restoration, better is the restoration performance achieved. Contrary to the regularizations on images, regularizations on blur kernels are more complicated because of the complex nature of the practical blurring processes. Depending on the sources of the blurriness of images, attempts have been made for regularizations on blur kernels. If the source of blurriness is motion of camera or objects, the blur kernel displays a trajectory with sharp edges, and can be modelled as a function of sparse support [5], [6], [35] or of a small total variation [14]. On the other hand, if the source is due to out-of-focus in camera parameter, then a Gaussian type of smoothing is usually used to model the blurriness [36]. A learning-based approach is also proposed to derive a blur kernel from an ensemble of training data [37]. Recently, Ji and Wang [38] analyzed the robustness issue of blind image restoration, and they reported that sometimes few errors on the estimated blind kernels can cause significant errors in the image reconstruction process. They introduced a model that can explicitly take into account the blur kernel estimation error in the regularization process in order to obtain restoration results that are robust to modelling errors of blur kernels.

TABLE I
DIFFERENT DEBLURRING PROBLEMS AND RELATED WORKS.

	Spatially invariant	Spatially variant
blind	[3], [4], [5], [6], [7], [13], [14], [16], [17], [18], [26], [30]	[2], [1], [25], [27], [29], [33]
non-blind	Wiener filter, RL method and [10], [15], [11], [21], [19], [8], [31], [9], [20].	[28], [32]

III. PROBLEM FORMULATION

Hereafter, we assume the blur kernel as a separable (block) circular convolution kernel. This assumption simplifies the derivations, brings computational efficiency (by employing two 1-D convolutions in place of a 2-D convolution), and maintains a generality in the proposed method.

We use the following model for blind image restoration:

$$\begin{cases} \min_{X, H_1, H_2} \|Y - H_1 X H_2^T\|_F^2 + R(X) + \mu \sum_{i,j} |\alpha_i \beta_j|, \\ H_1 = \sum_i \alpha_i D_i \\ H_2 = \sum_j \beta_j \hat{D}_j, \end{cases} \quad (4)$$

where H_1 and H_2^T are (block) circulant matrices along the columns and rows of X , respectively; D_i and \hat{D}_j are circulant

matrices representing basic blurring patterns for H_1 and H_2 , respectively. The sum of each row of H_i is normalized to 1, so that X and Y have the same mean. Let $\mathbf{1}$ be an $n \times n$ image of all entries equal to 1 and let $h_{i,k}$ be the k -column in H_i . Then, the mean of $H_1 \mathbf{1} H_2^T$ is 1, as

$$\frac{1}{n^2} \sum_{k,l=1}^n h_{1,k}^T \mathbf{1}_{k,l} h_{2,l} = 1. \quad (5)$$

We can make the dictionary for blur kernel explicit by taking *vec* operation on matrices in the objective function in Equation (4) to obtain

$$\min_{x,H} \|y - Hx\|^2 + R(x) + \sum_{i,j} |\alpha_i \beta_j|, \quad (6)$$

where

$$H = H_2 \otimes H_1 \quad (7)$$

$$= \sum_j \beta_j \tilde{D}_j \otimes \sum_i \alpha_i D_i \quad (8)$$

$$= \sum_{i,j} \alpha_i \beta_j (\tilde{D}_j \otimes D_i). \quad (9)$$

The term $\sum_{i,j} |\alpha_i \beta_j|$ in Equation (4) implies that the blur kernel H is sparse with respect to the tensor dictionary, formed by a stack of matrix $\tilde{D}_j \otimes D_i$. H is in the vector space spanned by atoms $\tilde{D}_j \otimes D_i$, which are formed by applying the Kronecker product of the two circulant matrices, \tilde{D}_j and D_i . Note that a circulant matrix can be constructed from a 1-D sequence and the circulant matrix maintains the structure for convolution that can be efficiently implemented in frequency domain.

A. Vector Space of Blur Kernel

Since a blur kernel is usually a low-pass filter due to incorrect setting of camera parameters, we construct the vector space for the blur kernel H by using low-pass circular patterns. We use Gaussian filters as our basis elements to compose the vector space for H because Gaussian filters are not only separable, but also the most prevailing blurring operator for out-of-focus blurring distortions.

One-dimensional Gaussian functions of various standard deviations from σ_1 to σ_N are uniformly quantized to obtain N discrete sequences. Each sequence is then normalized so that the sum of the sequence is equal to 1. The normalized sequence derived from the standard deviation σ_i is then used to specify and generate the circulant matrix G_i . Let

$$H_1 = \sum_{i=1}^N \alpha_i G_i, \quad (10)$$

and

$$H_2 = \sum_{i=1}^N \beta_i G_j. \quad (11)$$

By using vector space representation, the circulant matrices H_1 and H_2 can be characterized by their respective coefficient

vectors $\alpha = [\alpha_i]^T$ and $\beta = [\beta_i]^T$. Since the sums of each row of G_i and H_i have been normalized, we have

$$\sum_{i=1}^N \alpha_i = \sum_{i=1}^N \beta_i = 1. \quad (12)$$

Substituting Equations (10) and (11) into the first term of the objective function in problem (4), we obtain

$$\begin{aligned} Y &= H_1 X H_2^T + N \\ &= \sum_{i,j=1}^N \alpha_i \beta_j G_i X G_j^T + N, \end{aligned} \quad (13)$$

where $G_i X G_j^T$ is the blurred image of X , which is horizontally blurred by 1-D Gaussian of standard deviation σ_j and vertically blurred by that of standard deviation σ_i ; N is the noise.

The blur kernel H is a linear combination of basic pattern $G_j \otimes G_i$ as

$$H = H_2 \otimes H_1 \quad (14)$$

$$= \sum_{j=1}^N \beta_j G_j \otimes \sum_{i=1}^N \alpha_i G_i \quad (15)$$

$$= \sum_{i,j=1}^N \alpha_i \beta_j (G_j \otimes G_i). \quad (16)$$

As each $G_j \otimes G_i$ matrix is of dimension $n^2 \times n^2$, H is an $n^2 \times n^2$ matrix in the vector (sub)space of dimension N^2 , spanned by the N^2 matrices $G_j \otimes G_i$.

IV. NUMERICAL ALGORITHM

To estimate the blur kernel H , estimation of its respective coefficients α and β in the vector space is done, where α and β denote the vectors of α_i and β_j respectively. The proposed image restoration problem can now be re-expressed as

$$\begin{cases} \min_{X,\alpha,\beta} \|Y - \sum_{i,j=1}^N \alpha_i \beta_j G_i X G_j^T\|_F^2 + R(X) \\ \quad + \mu \sum_{i,j} |\alpha_i \beta_j|, \\ \sum_i \alpha_i = 1 \\ \sum_i \beta_i = 1. \end{cases} \quad (17)$$

The problem can be solved by using a relaxation approach that alternatively estimate X and (α, β) considering the other variable fixed. If (α, β) is fixed, X can be obtained by solving the following sub-problem:

$$\min_X \|Y - \sum_{i,j=1}^N \alpha_i \beta_j G_i X G_j^T\|_F^2 + R(X). \quad (18)$$

Meanwhile, if X is fixed, (α, β) can be estimated by solving the following sub-problem:

$$\begin{cases} \min_{\alpha,\beta} \|Y - \sum_{i,j=1}^N \alpha_i \beta_j G_i X G_j^T\|_F^2 + \mu \sum_{i,j} |\alpha_i \beta_j| \\ \sum_i \alpha_i = 1 \\ \sum_i \beta_i = 1. \end{cases} \quad (19)$$

If the regularization $R(X)$ is a convex function of X , then sub-problem (18) will be convex, too. Since sub-problems (18) and (19) are both convex, the convergence of alternative

approach to minimizers of X and (α, β) can be ensured. The algorithms to derive the minimizers of the above sub-problems are provided in the following subsections.

A. Estimation of Image

The algorithm for the sub-problem (18) can estimate image X . By choosing the regularization on X to be its total variation, sub-problem (18) becomes

$$\min_X \|Y - \sum_{i,j=1}^N \alpha_i \alpha_j G_i X G_j^T\|_F^2 + \delta(\|D_1 x\|_1 + \|D_2 x\|_1), \quad (20)$$

where x is a vector form of X , $D_1 x$ and $D_2 x$ denote the vectors of the first-order discrete horizontal difference and vertical difference at each pixel of X , respectively; δ is a Lagrangian parameter.

Based on the work of Wang et al.[19], we use a variable splitting technique to estimate the image. We replace $\|D_1 x\|_1$ by $\|v_1\|_1 + \frac{\gamma}{2}\|D_1 x - v_1\|_2^2$ and $\|D_2 x\|_1$ with $\|v_2\|_1 + \frac{\gamma}{2}\|D_2 x - v_2\|_2^2$, by introducing new variable vectors v_1 and v_2 . As $\gamma \rightarrow \infty$, it is clear that

$$\|v_i\|_1 + \frac{\gamma}{2}\|D_i x - v_i\|_2^2 \rightarrow \|D_i x\|_1, \quad (21)$$

for $i = 1, 2$. Sub-problem (18) can now be re-written as

$$\min_{X, v} \|Y - \sum_{i,j=1}^N \alpha_i \beta_j G_i X G_j^T\|_F^2 + \delta \sum_{i=1}^2 (\|v_i\|_1 + \frac{\gamma}{2}\|v_i - D_i x\|_2^2), \quad (22)$$

where $v = [v_1^T v_2^T]^T$. The solution of the above equation converges to that of sub-problem (18) as $\gamma \rightarrow \infty$. The variable splitting technique is extremely efficient, because when either of the two variables in Equation (22) is fixed, minimizing the equation with respect to the other has a closed-form solution. The overall convergence of this minimization algorithm is well analyzed in [19].

For a fixed X , variables v_1 and v_2 can be derived separately by solving

$$\min_{v_i} \|v_i\|_1 + \frac{\gamma}{2}\|v_i - D_i x\|_2^2. \quad (23)$$

The l_1 -minimizer is given by the following shrinkage formula:

$$v_i = \max \left\{ |D_i x| - \frac{1}{\gamma}, 0 \right\} \text{sign}(D_i x), \quad (24)$$

where all operations are done component-wise. On the other hand, for a fixed v , the minimizer X can be derived by

$$\min_x \|y - \sum_{i,j=1}^N \alpha_i \beta_j (G_j \otimes G_i) x\|_F^2 + \frac{\delta \gamma}{2} \sum_{i=1}^2 \|v_i - D_i x\|_2^2, \quad (25)$$

where the first term is obtained by taking vec operation on the first term of sub-problem (18). Let $A = \sum_{i,j=1}^N \alpha_i \beta_j (G_j \otimes G_i)$. Taking the partial derivative of Equation (25) with respect to x and setting the resultant to zero, we obtain the following equation for the minimizer x :

$$\left(\sum_{i=1}^2 D_i^T D_i + \frac{2}{\delta \gamma} A^T A \right) x = \sum_{i=1}^2 D_i^T v_i + \frac{2}{\delta \gamma} A^T y. \quad (26)$$

If X is an $n \times n$ matrix, then the size of A will be $n^2 \times n^2$, which becomes cumbersome for direct computation of Equation (26) as we need to inverse a huge matrix. Thus, it is suggested in [19] to solve it by using the FFT.

If we explore the (block) circulant matrix structure of D_1 , D_2 , and each term in A [28], all matrix multiplications in Equation (26) are convolution operations. Let $\mathcal{F}_1(C)$ denote the 1-D Fourier transform of the generating sequence of circulant matrix C . Through convolution theorem of Fourier transform, Equation (26) can be written as

$$x = \mathcal{F}_1^{-1} \left(\frac{\sum_{i=1}^2 (\overline{\mathcal{F}_1(D_i)} \circ \mathcal{F}_1(v_i)) + \frac{2}{\delta \gamma} \overline{\mathcal{F}_1(A)} \circ \mathcal{F}_1(y)}{\sum_{i=1}^2 (\overline{\mathcal{F}_1(D_i)} \circ \mathcal{F}_1(D_i)) + \frac{2}{\delta \gamma} \overline{\mathcal{F}_1(A)} \circ \mathcal{F}_1(A)} \right), \quad (27)$$

where \mathcal{F}_1 and \mathcal{F}_1^{-1} denote the forward and inverse 1-D Fourier transform, respectively, \circ denotes component-wise multiplication, and $\overline{\mathcal{F}_1(C)}$ is the complex conjugate of $\mathcal{F}_1(C)$. Since only the variables v_1 and v_2 of Equation (27) are changed during each iteration, computation loads can be reduced by computing the FFT of all the other variables in advance.

B. Estimation of the Blur Kernel

If we let $Z_{i,j} = G_i X G_j^T$, the sub-problem (19) can be expressed as

$$\begin{cases} \min_{\alpha, \beta} \|Y - \sum_{i,j=1}^N \alpha_i \beta_j Z_{i,j}\|_F^2 + \mu \sum_{i,j} |\alpha_i \beta_j| \\ \sum_i \alpha_i = 1 \\ \sum_i \beta_i = 1. \end{cases} \quad (28)$$

Let U be the rank one matrix $\alpha \beta^T$, where α and β are vectors of α_i and β_i , respectively; and let $u = \text{vec}(U)$. We have $\sum_{i,j} |\alpha_i \beta_j| = \|u\|_1$. In addition, If vec is applied on the first term of problem (28), we obtain

$$\|y - Zu\|_2^2, \quad (29)$$

where $y = \text{vec}(Y)$ and Z is a matrix with columns corresponding to the vectors derived by $\text{vec}(Z_{i,j})$. Problem (28) can now be re-expressed as

$$\begin{cases} \min_{\alpha, \beta} \frac{1}{2} \|y - Zu\|_2^2 + \mu \|u\|_1 \\ u = \text{vec}(U) \\ U = \alpha \beta^T // U \text{ is a rank one matrix.} \\ \sum_i \alpha_i = 1 \\ \sum_i \beta_i = 1. \end{cases} \quad (30)$$

First, all constraints of the above problem are ignored, and the proximal gradient method is applied to derive u by solving the objective function. Then, the constraints are imposed on u so that the resultant matrix $U = \text{vec}^{-1}(u)$ becomes a rank one matrix with $\sum_i \alpha_i = \sum_i \beta_i = 1$.

The first term in the objective of problem (30) is differentiable and the second term is indifferentiable. The solver of the proximal gradient method is thus

$$u^{k+1} := \text{prox}_{\mu^k \|\cdot\|_1} (u^k - \mu^k \nabla f(u^k)), \quad (31)$$

where

$$\nabla f(u^k) = Z^T Z u^k - Z^T y, \quad (32)$$

and $\text{prox}_{\mu^k \|\cdot\|_1}$ is soft thresholding and μ^k is a step size. The step size μ^k is determined by backtracking line search process. The proximal gradient method to derive initial u is detailed in the following.

given: u^k , μ^0 , and parameter $\beta \in (0, 1)$.

Let $\mu := \mu^0$.

repeat

1. Let $z := \text{prox}_{\mu \|\cdot\|_1}(u^k - \mu \nabla f(u^k))$.
2. **break if** $f(z) \leq \hat{f}_\mu(z, u^k)$. //defined in Equation (33)
3. Update $\mu := \beta\mu$. Go to 1.

return $\mu := \mu^0$, $u^k := z$

The typical value for the line search parameter β is set at 0.5, and the \hat{f}_μ in line 2 is the stopping criterion given as

$$\hat{f}_\mu(x, y) = f(y) + \nabla f(y)^T(x - y) + \frac{1}{2\mu} \|x - y\|_2^2, \quad (33)$$

with $\mu > 0$. This proximal gradient method needs to be evaluated many times in the minimization process until the optimal u is reached (in line **return**).

Let \hat{u} be the result of the proximal gradient method. Then, we apply vec^{-1} on \hat{u} to obtain the matrix \hat{U} . Since the objective matrix U is a rank one matrix, singular value decomposition is applied on \hat{U} to obtain

$$\hat{U} = S\Sigma V^T, \quad (34)$$

where the singular values are arranged in a non-increasing order¹. The optimum rank one approximation of \hat{U} is the matrix of $\sigma s v^T$, where s is the first column in S , v^T is the first row in V^T and σ is the largest eigenvalue in Σ . Since

$$U = \sigma s v^T = \alpha \beta^T, \quad (35)$$

we can factor $\sigma = ab$ such that

$$\sum_i \alpha_i = a \sum_i s_i = 1 \quad (36)$$

and

$$\sum_i \beta_i = b \sum_i v_i = 1. \quad (37)$$

Multiplying both sides of the last equation by a , we have

$$a = \sigma \sum_i v_i. \quad (38)$$

Note that if we know $H_1 = H_2$, then $\alpha = \beta$ and $a = b = \sqrt{\sigma}$.

C. Algorithm

Figure 1 displays an overview of the proposed alternative optimization approach, with the stepwise algorithm given in the following table.

¹If $H_1 = H_2$ (the horizontal blurring and the vertical blurring are performed by the same kernel), we can project \hat{U} into the vector space of symmetric matrix to obtain its projection $\frac{\hat{U} + \hat{U}^T}{2}$, which is then a rank one symmetric matrix.

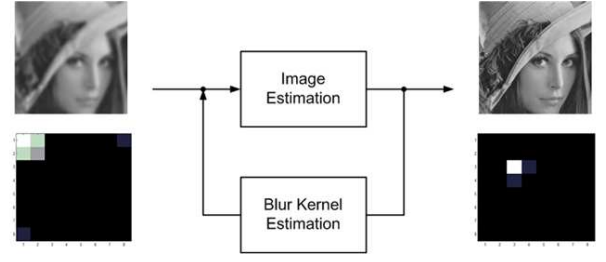


Fig. 1. Overview of the proposed algorithm. The top left subimage is the observed blurred image. The bottom left subimage is formed from the coefficients of the initial blur kernel, with the horizontal and vertical axes corresponding to the initial α and β vector, respectively. Image and blur kernel are iteratively refined until the convergence is reached, as shown in the right subimages.

Input: Blurry image Y ; initial α and β ; parameters $\gamma < p$, where p is a given constant; δ in Equation (20); a dictionary of N^2 basic patterns; and the maximum number of iterations M .

Let $X = Y$. // Image is initialized as Y .

1. If $\gamma < p$, then

repeat

2. Estimate image $X = \text{vec}^{-1}(x)$ according to Equation (27).

3. Estimate α and β by solving problem (30):

3.1. Update u by proximal gradient method, based on Equation (31).

3.2. Derive the rank one approximation U of $\text{vec}^{-1}(u)$.

3.3. Determine α and β from SVD of U , based on Equation (38).

4. **break if** maximum iteration number M is reached.

5. Increase γ . Go to 1.

6. Derive H from the coefficients α and β , based on Equation (9).

return Image X and blur kernel H .

The outer loop, composed of steps 1 and 5, of this algorithm increase the parameter γ for the variable splitting method. Starting with a small γ value, our algorithm gradually increase its value to reach a given constant p . The inner loop (including steps 2, 3, and 4) alternatively estimate image X and the blur kernel coefficients α and β . In each step of estimation in the inner loop, the minimum of a convex function is found; therefore, the inner loop always show convergence. Without degrading the overall performance of our algorithm, we set a maximum number of iterations to enforce the inner loop to stop before it reaches the convergence.

Because of non-convexity of the blind image restoration problem, initial guess of H is surely important for the final restoration result. Empirically, the current method can yield a good restoration image if the prior information about the blur kernel can be well estimated. Degrading the initial guess could result in a slow-paced convergence. Since H is assumed to be sparse, if the number of atoms in a dictionary is large enough, a good initial guess of H can be the one sparse function that approximates H by one atom in the dictionary. For an image

subjected to Gaussian-type out-of-focus blurriness, there are algorithms that can reliably estimate the dominating Gaussian blur kernel on the image. For other dictionaries, where it is difficult to identify the dominating atom, the initial guessing of H cannot be easily identified, an exhaustive approach based on trying on each atom as the initial guess of H can be applied.

V. EXPERIMENTAL RESULTS

Here we intend to show that our method can be effectively performed on different images with degradations by various blur kernels in both noise-less and noisy environments.

A. Dictionary Design and Initial Guess of H :

In this part, a detailed description of our dictionary design has been provided. We chose N scaled 1-D Gaussian function, $G(\sigma)$, as the building block for the 2D-atoms of our dictionary. The dictionary has N^2 atoms, each is of the form $G(\sigma_i) \otimes G(\sigma_j)$, with $i, j = 1, \dots, N$. Increasing N can increase the restoration performance, but also increase the computational complexity at the same time. To achieve an optimal balance between performance and computational efficiency, the value of N is empirically determined to be eight. The standard deviations of 1-D Gaussian functions corresponding to the eight number are set from 0.5, 1, \dots , 3.5, 4. For memory consideration, we processed our method by blocks, where an image is divided into blocks of size 32×32 . Instead of processing each 32×32 block directly, to avoid the boundary artefact, we processed on a larger block (called processing block) of size 96×96 that embeds a 32×32 block in the center and then took the center 32×32 block in the result. Therefore, our dictionary has 64 2-D patterns, each of dimension $96^2 \times 96^2$.

We use the all-focused method in [25] to estimate the out-of-focus blur on an image. The blurriness of step edges is modelled as the convolution of a 2-D Gaussian function, approximating the point spread function of a camera. From the horizontal and vertical blurriness, the standard deviation of the Gaussian function is derived. Let σ_{i_0} and σ_{j_0} be the estimated horizontal and vertical standard deviations, respectively. Since our dictionary has only 64 atoms, this number is not large enough to approximate a blur kernel by one sparse function of atom, we thus used the basis pursuit denoising algorithm (BPDN) [23] to derive the approximation of $G(\sigma_{i_0}) \otimes G(\sigma_{j_0})$. From Equations (10) and (11), the following optimization problem is solved by BPDN:

$$\begin{cases} \min \|\alpha\|_1 + \|\beta\|_1 \\ \|G(\sigma_{i_0}) - \sum_{i=1}^N \alpha_i G(\sigma_i)\|_2 \leq \tau \\ \|G(\sigma_{j_0}) - \sum_{i=1}^N \beta_i G(\sigma_i)\|_2 \leq \tau, \end{cases} \quad (39)$$

where τ is given as the error bound. Let the index set of non-zero coefficients of α be I_1 and that of non-zero coefficients of β be I_2 . Then, the initial guess of H is $H_2^0 \otimes H_1^0$, where $H_1^0 = \sum_{i \in I_1} \alpha_i G(\sigma_i)$ and $H_2^0 = \sum_{i \in I_2} \beta_i G(\sigma_i)$.

B. Comparisons and Algorithm Parameters:

We further compare our results with two other deblurring methods, viz. the matlab built-in function *deconvblind* and the method proposed by Krishnan *et al.* [18]. The *deconvblind* deconvolves a blurred image by using the maximum likelihood

TABLE II
TH PSFS OF BLUR KERNELS USED FOR OUR COMPARISONS.

kernel	PSF
1	disk with radius=5
2	$H = h_0 h_0^T$, $h_0^T = [1 \ 9 \ 36 \ 84 \ 126 \ 126 \ 84 \ 36 \ 9 \ 1]$
3	Gaussian with $\sigma=2.6$
4	$H=1/(1+x_1^2+x_2^2)$, $x_1, x_2=-7, \dots, 7$

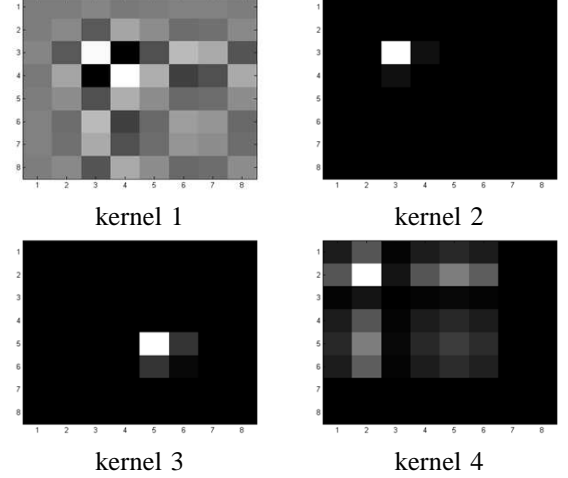


Fig. 2. Representing the kernels in Table II with respect to our dictionary. The sub-figures are the absolute values of coefficients $\alpha\beta^T$. Kernels 2 and 3 are sparse.

algorithm. The *deconvblind* has several optional parameters, e.g. number of iterations. *deconvblind* is quite fast, but the deblurred results in terms of PSNR as well as visual quality are often unsatisfactory, even with several iterations. On the other hands, Krishnan's method has many manually selected parameters. The two parameters that we chose are different from their default values. The chosen parameters are λ (the regularizing parameter), which ranged from 60 to 100, and the iteration number, which is set to 20.

We used the four 512×512 grayscale images, *Lena*(Img01), *Cameraman*(Img02), *House*(Img03) and *Mandrill*(Img04), and four blur kernels in [39] as our benchmark for the first two experiments. In total, we have 16 blurred images. The point spread function (PSF) of each blur kernel H is normalized so that $\sum_i h_i = 1$. The numerical values of the kernels are given in Table II. Figure 2 shows the maps of the absolute values of coefficients $\alpha\beta^T$, defined in Equations (30) and (35), for each of the four blur kernels. The darker a pixel is, smaller the value of the pixel has. As shown in the figure, kernels 2 and 3 are sparse with respect to our dictionary, and Kernel 4 is almost sparse.

To make a quantitative comparison, we use the peak signal-to-noise ratio (PSNR) and sum-of-squared differences (SSD) to measure the accuracy of the deblurred image and the estimated PSF, respectively. The experimental results for all the compared methods are derived based on the same initial kernel, as described in Part A of this section, and the same number of iterations (20 runs for all cases). The step size used in our proximal gradient method is chosen with $\mu \in [5e^{-11}, 5e^{-13}]$ and the value of γ is from 1 to 100 (the

TABLE III
COMPARISON OF THE OUTPUT PSNR [dB] AND SSD OF THE THREE DECONVOLUTION METHODS ON 16 IMAGES.

		Kernel 1		Kernel 2		Kernel 3		Kernel 4	
		PSNR	SSD	PSNR	SSD	PSNR	SSD	PSNR	SSD
Img01	deconvblind	26.54	0.0959	29.32	0.0319	27.07	0.0018	30.83	0.0283
	Krishnan et al.	22.42	0.0065	23.41	0.0087	24.05	0.0170	29.30	0.0027
	Our method	26.48	0.0948	28.56	0.0104	28.58	0.0004	31.20	0.0030
Img02	deconvblind	27.27	0.0959	31.18	0.0319	28.06	0.0023	32.29	0.0280
	Krishnan et al.	22.31	0.0112	28.61	0.0189	29.87	0.0237	33.01	0.0027
	Our method	26.95	0.0971	31.28	0.0104	29.35	0.0004	31.07	0.0030
Img03	deconvblind	31.94	0.0959	35.39	0.0319	32.57	0.0015	28.75	0.0289
	Krishnan et al.	25.09	0.0059	27.53	0.0122	28.02	0.0071	35.06	0.0043
	Our method	32.51	0.0963	35.54	0.0106	32.65	0.0004	34.70	0.0032
Img04	deconvblind	21.92	0.0958	25.32	0.0318	22.64	0.0024	27.65	0.028
	Krishnan et al.	19.79	0.0109	21.66	0.0044	19.18	0.0315	24.84	0.0071
	Our method	21.97	0.0955	24.57	0.0104	23.56	0.0004	29.24	0.0046

TABLE IV
COMPARISON OF THE OUTPUT PSNR [dB] OF THE THREE DECONVOLUTION METHODS WITH IMAGES STAINED BY 30 DB ADDITIVE WHITE GAUSSIAN NOISE.

Each with noise 30dB		Kernel 1		Kernel 2		Kernel 3		Kernel 4	
		PSNR	SSD	PSNR	SSD	PSNR	SSD	PSNR	SSD
Img01	deconvblind	24.56	0.1123	25.97	0.0319	19.67	0.0018	24.33	0.0858
	Krishnan et al.	22.32	0.0089	23.22	0.0088	23.86	0.0166	29.54	0.0055
	Our method	23.78	0.1022	26.51	0.0140	27.01	0.0008	28.11	0.0048
Img02	deconvblind	25.73	0.1121	28.20	0.0319	19.26	0.0023	25.20	0.0857
	Krishnan et al.	22.25	0.0119	28.43	0.0188	29.68	0.0247	33.51	0.0072
	Our method	25.63	0.1046	29.68	0.0148	29.00	0.0007	27.29	0.0043
Img03	deconvblind	24.93	0.1127	24.85	0.0319	22.10	0.0015	25.13	0.0858
	Krishnan et al.	24.98	0.0071	27.22	0.0022	27.72	0.0055	34.38	0.0045
	Our method	30.98	0.1029	31.86	0.0141	31.28	0.0007	30.56	0.0037
Img04	deconvblind	21.71	0.1119	23.41	0.0318	15.55	0.0024	24.34	0.0857
	Krishnan et al.	19.56	0.0160	21.62	0.0062	19.03	0.0329	25.24	0.0046
	Our method	21.06	0.1028	23.79	0.0135	21.03	0.0007	25.76	0.0049

value of p in our algorithm) and δ is between $[10^{-3}, 10^{-1}]$.

We performed the first experiment in a noise-less environment. The PSNR of the compared algorithms on all test images are shown in Table III. Out of the 16 test images, four best results are from *deconvblind*, three from Krishnan et al., and nine from our method. With respect to the SSD values on the estimated blur kernels, our method yields the best estimation in almost all cases. Figure 3 compares the visual quality of the deblurring results of all the methods.

We performed the second experiment in a noisy environment when a white Gaussian noise at a signal-to-noise ratio of 30 dB is added to the blurred images. The PSNRs and SSDs of all the compared methods of this experiment are shown in Table IV. To understand the robustness, for each kernel and each method, we calculated the average PSNR reduction of the four images (Img01, Img02, Img03, and Img04) by

$$\frac{1}{4} \sum_{i=1}^4 (\text{PSNR at 30 dB of image } i - \text{PSNR at noiseless of image } i), \quad (40)$$

and the results have been presented in Table V. Note that we have removed the comparison with the Krishnan et al.'s method in the table, because if we compared the PSNRs in Tables III and IV, the Krishnan et al.'s method would be 3-4 dB in average lower than our method. As shown in Table V, our method has a smaller PSNR reduction for each kernel than that of the *deconvblind*, indicating more robustness of our method in the restoration of images in a noisy environment. Table VI



Fig. 4. Highlighted the cheek of bridegroom in Figure 5. (a) Result of *deconvblind* and (b) our method. Sparkle artefact can be found in (a).

compares the average PSNR differences of our method and the *deconvblind* for each kernel on all test images in both noise-less and noisy environments. Except for the noise-less and Kernel 2 case, our method outperforms *deconvblind*. The sparsity demonstrates its robustness in the cases of 30 dB and Kernels 2 and 3, where our method achieves high PSNR gain over *deconvblind*.

Finally, figures 5 and 6 visually compare the deblurring results of color-images, taken in real-life. The photographs contain complex structures and different degrees of blurriness. As shown in Figure 5, the faces of the dolls are clearly restored by all methods. However, if we zoomed in the cheek of the bridegroom, as shown in Figure 4, the cheek from our result is smooth while that from the *deconvblind* has some sparkles in it. Furthermore, figure 6 shows that our method successfully enhanced the sharp edges.

TABLE V

COMPARISON OF THE AVERAGE PSNR REDUCTION FOR EACH KERNEL, ACCORDING TO EQUATION (40).

	Kernel 1	Kernel 2	Kernel 3	Kernel 4
deconvblind	-2.6850	-4.6950	-8.4400	-5.1300
Our method	-1.6150	-2.0275	-1.4550	-3.6225

TABLE VI

THE AVERAGE PSNR GAIN OF KERNELS OF OUR METHOD OVER *deconvblind* IN NOISE-LESS AND NOISY ENVIRONMENT. NOTICE THE GAINS OF KERNELS 2, 3, AND 4 FOR NOISY ENVIRONMENT, WHERE KERNELS 2 AND 3 CAN BE SPARSELY REPRESENTED BY OUR DICTIONARY AND KERNEL 4 IS ALMOST SPARSE.

	Kernel 1	Kernel 2	Kernel 3	Kernel 4
noiseless	0.06	-0.3150	0.95	1.6724
30 dB noise	1.13	2.3525	7.935	3.18

VI. CONCLUSIONS

Regularization on an unknown blur kernel determines the performance of the blind image restoration problem. In the current paper, we have proposed a novel approach to construct regularization by modelling a blur kernel as a sparse representation of a tensor dictionary, where the dictionary is composed of basic 2-D pattern. Since the dictionary approach has the freedom to be customized for various applications, our approach can be used to connect various regularizations that have been imposed on blur kernels in different applications. As a demonstration, we construct a dictionary with atoms formed by the Kronecker product of two 1-D scaled Gaussian functions and show that this dictionary can effectively restore images blurred by the mixed Gaussian types of blur kernels. We also demonstrate that our approach can be efficiently solved by using the variable splitting method for image estimation and proximal gradient method for blur kernel estimation. Furthermore, we compare the performance of our method with some state-of-the-art methods for various sets of images and blur kernels. In most cases, our method derives the best image (in terms of PSNR) as well as blur kernel estimation. An interesting direction for further study is to incorporate a learning procedure to our approach for various applications.

ACKNOWLEDGMENT

The authors would like to thank...

REFERENCES

- [1] A. Chakrabarti, T. Zickler, and W. Freeman, "Analyzing spatially-varying blur," *CVPR*, pp. 2512-2519, Jun. 2010.
- [2] J. Jia, "Single image motion deblurring using transparency," *CVPR*, pp. 1-8, Jun. 2007.
- [3] J. Pan, Z. Su, "Fast l^0 -regularized kernel estimation for robust motion deblurring," *IEEE Sig. Proc. Lett.*, Vol. 20 Issue 9, pp. 841-844, Sep. 2013.
- [4] T.F. Chan, C.K. Wong, "Total variation blind deconvolution," *IEEE Trans. on Img. Proc.*, Vol. 7, Issue 3, pp. 370-375, Mar. 1998.
- [5] R. Fergus, B. Singh, A. Hertzmann, S. Roweis, and W. Freeman, "Removing camera shake from a single image," *ACM Trans. on Graphics*, Vol. 25 Issue 3, pp. 787-794, Jul. 2006.
- [6] Q. Shan, J. Jia, and A. Agarwala, "High-quality motion deblurring from a single image," *ACM Trans. on Graphics*, Vol. 27, Issue 3, No. 73, Aug. 2008.
- [7] A. Levin, Y. Weiss, Y. Durand, and W. Freeman, "Understanding and evaluating blind deconvolution algorithms," *CVPR*, pp. 1964-1971, Jun. 2009.
- [8] J. Bect, L. Blanc-Feraud, G. Aubert, and A. Chambolle, "A l^1 -unified variational framework for image restoration," *ECCV*, pp.1 - 13, 2004.
- [9] M. Figueiredo, R. Nowak, "A bound optimization approach to wavelet-based image deconvolution," *IEEE ICIP*, 2005.
- [10] L. Yuan, J. Sun, L. Quan, and H.Y. Shum, "Image deblurring with blurred/noisy image pairs," *ACM Trans. on Graphics*, Vol. 26, Issue 3, No. 1, Jul. 2007.
- [11] A. Rav-Acha, S. Peleg, "Two motion-blurred images are better than one," *Pattern Recognition Letters*, Vol. 26, Issue 3, pp. 311-317, Feb. 2005.
- [12] H. Yin, I. Hussain, "Blind source separation and genetic algorithm for image restoration," *ICAST* pp. 167-172, 2006.
- [13] H. Zhang, J. Yang, Y. Zhang, and T.S. Huang, "Sparse representation based blind image deblurring," *Proc. IEEE Conf. Multimedia Expo.*, pp. 1-6, 2011.
- [14] J. Money, S.H. Kang, "Total variation minimizing blind deconvolution with shock filter reference," *Image and Vision Computing*, Vol. 26, Issue 2, pp. 302314, Feb. 2008.
- [15] S. Zhuo, D. Guo, and T. Sim, "Robust flash deblurring," *CVPR*, pp. 2440-2447, 2010.
- [16] S. Babacan, R. Molina, and A. Katsaggelos, "Variational bayesian blind deconvolution using a total variation prior," *IEEE Trans. on Img. Proc.*, Vol. 18, Issue 1, pp. 12-26, Jan. 2009.
- [17] S. Cho, S. Lee, "Fast motion deblurring," *ACM Trans. on Graphics*, Vol. 28, No. 5, pp. 145:1-145:8, Dec. 2009.
- [18] D. Krishnan, T. Tay, and R. Fergus, "Blind deconvolution using a normalized sparsity measure," *CVPR*, pp. 233-240, Jun. 2011.
- [19] Y. Wang, J. Yang, W. Yin, and Y. Zhang, "A new alternating minimization algorithm for total variation image reconstruction," *SIAM J. Imag. Sci.*, Vol. 1, Issue 3, pp. 248-272, Aug. 2008.
- [20] D. Krishnan, R. Fergus, "Fast image deconvolution using hyper-laplacian priors," *NIPS*, 2009.
- [21] M. Almeida, L. Almeida, "Blind and semi-blind deblurring of natural images," *IEEE Trans. on Img. Proc.*, Vol. 19, Issue 1, pp. 36-52, Jan. 2010.
- [22] S. Osher, L. I. Rudin, "Feature-oriented image enhancement using shock filters," *SIAM J. Num. Anal.*, Vol. 27, Issue 4, pp. 919-940, Aug. 1990.
- [23] E. van den Berg and M. P. Friedlander, "SPGL1: A solver for large-scale sparse reconstruction," <http://www.cs.ubc.ca/labs/sci/spgl1>, Jun. 2007.
- [24] N. Parikh and S. Boyd, "Proximal algorithms," *Found. Trends Optim.*, Vol. 1, No. 3, pp. 123231, 2013.
- [25] W. Zhang, W.K. Cham, "Single-image refocusing and defocusing," *IEEE Trans. on Img. Proc.*, Vol. 21, Issue 2, pp. 873-882, Feb. 2012.
- [26] J. Cai, H. Ji, C. Liu, and Z. Shen, "Blind motion deblurring from a single image using sparse approximation," *CVPR*, 2009.
- [27] A. Levin, R. Fergus, F. Durand, and W. Freeman, "Image and depth from a conventional camera with a coded aperture," *ACM SIGGRAPH*, 2007.
- [28] M.K. NG, R.H. Chan, and W.C. Tang, "A fast algorithm for deblurring model with Neumann boundary conditions," *SIAM J. Sci. Comput.*, Vol. 21, pp.851-866, 1999.
- [29] S. H. Chan, T. Q. Nguyen, "Single image spatially variant out-of-focus blur removal," *ICIP*, pp.11 - 14, 2011.
- [30] A. Zunino, F. Benvenuto, E. Armadillo, M. Berto, and E. Bozzo, "Iterative deconvolution and semiblind deconvolution methods in magnetic archaeological prospecting," *GEOPHYSICS*, Vol. 74, No. 4, pp.43 - 51, 2009.
- [31] Y. Li, F. Santosa, "A computational algorithm for minimizing total variation in image restoration," *IEEE Trans. on Img. Proc.*, Vol. 5, Issue 6, pp. 987-995, Jun. 1996.
- [32] J. Nagy, D. O'Leary, "Restoring images degraded by spatially variant blur," *SIAM J. Sci. Comput.*, Vol. 19 No. 4, pp.1063 - 1082, Jul. 1998.
- [33] M. Ozkan, A. Tekalp, and M. Sezan, "POCS-based restoration of space-varying blurred images," *IEEE Trans. on Img. Proc.*, Vol. 3, Issue 4, pp. 450-454, Jul. 1994.
- [34] E. Kee, S. Paris, S. Chen, J. Wang, "Modeling and removing spatially-varying optical blur," *IEEE ICCP*, pp. 1-8, 2011.
- [35] Y. Xu, X. Hu, L. Wang, and S. Peng, "Single image blind deblurring with image decomposition," *IEEE ICASSP*, 2012, pp. 929-932.
- [36] Y. You and M. Kaveh, "Blind image restoration by anisotropic regularization," *IEEE Trans. on Image Processing*, vol. 8, no. 3, pp. 396-407, 1999.
- [37] J. Miskin and D. J. C. MacKay, "Ensemble learning from blind image separation and deconvolution," *Adv. in Independent Component Analysis*, 2000.

- [38] H. Ji and K. Wang, "Robust image deblurring with an inaccurate blur kernel," *IEEE Trans. on Img. Proc.*, Vol. 21, Issue 4, pp. 1624-1634, Apr. 2012.
- [39] A. Danielyan, V. Katkovnik, and K. Egiazarian, "BM3D frames and variational image deblurring," *IEEE Trans. on Img. Proc.*, Vol. 21, Issue 4, pp. 1715-1728, Nov. 2011.



Fig. 3. Comparison of Deblurring results: (a) *House* with kernel 1; (b) *Cameramen* with kernel 2; (c) *Lena* with kernel 3; and (d) *Mandrill* with kernel 4. From the top row to the bottom one cropped fragments of images are present in the following order: original, blurred, reconstructed by *deconvblind*, Krishnan *et al.* and the current proposed method. Note that Krishnan's images (the fourth row) looks good, but they are often too sharp to have a high PSNR value.



Fig. 5. Results of real-life photographs. (a) Blurred image. (b) *deconvblind*. (c) Krishnan's result. (d) Our result.

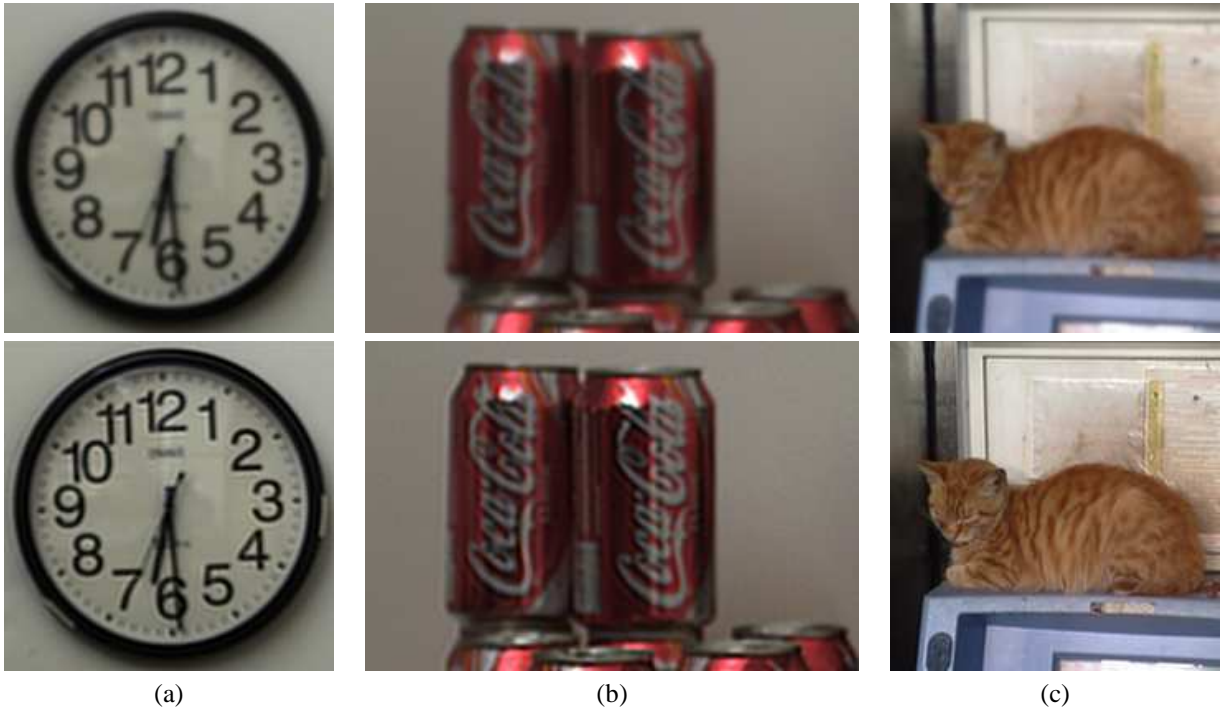


Fig. 6. Deblurring results of real photographs. Top row: blurred images. Second row: deblurring results of the proposed method.